DOCUMENT RESUME

ED 427 702 IR 019 253

AUTHOR Fenn, B.; Lennon, J.
TITLE Using a Recommender System and Hyperwave Attributes To Augment an Electronic Resource Library.
PUB DATE 1998-11-00
NOTE 8p.; In: WebNet 98 World Conference of the WWW, Internet, and Intranet Proceedings (3rd, Orlando, FL, November 7-12, 1998); see IR 019 231.
PUB TYPE Reports - Descriptive (141) -- Speeches/Meeting Papers (150)
EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Artificial Intelligence; Computer Software Development; Computer System Design; Cooperation; Electronic Libraries; Foreign Countries; Group Activities; Higher Education; *Hypermedia; Information Management; *Information Systems; Multimedia Materials; Online Systems; *Resource Materials; *User Needs (Information); Users (Information); *World Wide Web
IDENTIFIERS Client Server Computing Systems; *Information Value; Interest Groups; Links (Indexing); Prototypes; University of Auckland (New Zealand); *Web Sites

ABSTRACT

There has been increasing interest over the past few years in systems that help users exchange recommendations about World Wide Web documents. Programs have ranged from those that rely totally on user pre-selection, to others that are based on artificial intelligence. This paper proposes a system that falls between these two extremes, providing distinct advantages over other existing systems since it utilizes the structure and attributes associated with Hyperwave documents. This recommender system, which also ensures integrity of external links through its underlying server environment, can enhance the knowledge base of learning associations and other intranet groups. Following an introduction to the context of the project and user participation, the paper briefly describes the features of the Hyperwave server, a document management system for large quantities of multimedia data that can be spread over multiple remote servers. The Simple Recommender System (SRS), prototyped at the University of Auckland (New Zealand), is then detailed, including client-server arrangements and features that support making and receiving recommendations. The issue of maintaining referential integrity within a library of recommendations is considered. (Author/AEF)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

ED 427 702

Using a Recommender System and Hyperwave Attributes to Augment an Electronic Resource Library

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
☐ This document has been reproduced as received from the person or organization originating it.
☐ Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

B. Fenn
Aotea Interactive Media
Aotearoa New Zealand
barry@aotea.co.nz

J. Lennon
HyperMedia Unit, Computer Science Department
The University of Auckland, New Zealand
jennifer@cs.auckland.ac.nz

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
G.H. Marks
TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)"

Abstract: There has been increasing interest over the past few years in systems that help users exchange recommendations about web documents. Programs have ranged from those that rely totally on user pre-selection, to others that are based on artificial intelligence. We propose a system that falls between these two extremes, which has distinct advantages over other existing systems since it utilises the structure and attributes associated with Hyperwave documents. This recommender system, which also ensures integrity of external links through its underlying server environment, can enhance the knowledge base of learning associations and other intranet groups.

1 Introduction

Online recommender systems are a relatively new idea, although several innovative projects were described in detail within a special issue of Communications of the ACM [Communications, 1997]. They offer a shared knowledge base of empowerment for online collaboration; to which our intended development adds several unique features. In addition to generating personalised pages of new recommendations, the system will provide users with a historical record of their past and present recommendations to search, review, or rearrange. Each user will be able to review the results of their own archive, but since the recommendation links are to be stored in Hyperwave's structured database, users can also browse and query an anonymous display of an entire group's online history of selective media collaboration. This shared resource can be 'seeded' with a carefully selected set of links at the time a group of some kind is formed, giving new users a basis upon which to begin interacting.

Since this system relies heavily on certain characteristics of the Hyperwave server (developed by Professor Hermann Maurer and his colleagues in Graz, Austria) we will describe these capabilities briefly in Section 2.

1.1 The Context of the Project

The recommender system we describe here evolved from the more general concept of a system that would produce a personalised electronic 'newspaper' for members of the web community at large [Fenn & Shearer, 1996]. In this study we focus on tighter-knit communities such as those in intranet contexts, e.g., those in banking consortiums, medical groups, and education systems.

1.2 User Participation

Our initial approach assumes that all members of the participating group(s) have the same basic aims and objectives, with strong personal incentives toward collaborative cooperation. This ensures that users are sufficiently motivated to quite regularly pass on recommendations for the good of the group as a whole.

IR019253

BEST COPY AVAILABLE

2 Hyperwave

It is appropriate to include a brief summary of some of Hyperwave's main features, as these relate to the capabilities of the recommender system that we describe here. More detailed information can be obtained in both book form [Maurer, 1996] and online [http://www.iicm.edu/hw_mm].

Hyperwave was designed and developed as a document management system for very large quantities of multimedia data that can be spread over multiple remote servers. Since efficient maintenance of such massive collections of data is a critical issue, a structured approach was pursued right from the beginning (of what was originally called Hyper-G) [Andrews et al., 1994, Andrews et al., 1995, Flohr, 1995].

The Hyperwave server consists of a set of object-oriented databases. Hence, it supports a range of useful features such as assigning attributes to objects, full-text indexing, and searching (which, in this case, can be confined to a limited scope). The server also maintains a hierarchy of user and group identities.

A variety of attributes can be assigned to each document:

- Multiple, searchable keywords
- Language (with support for multilingual clusters)
- Display properties; such as sequence within a hierarchy
- Read and write permissions for both groups and individuals
- Direct debit, from the user's account, of the cost of viewing a document
- Setting of a time period during which the document is visible to other users
- Any other custom or system-specific attributes

Since attributes are not stored within the documents themselves (as is the case with most web servers), but in a separate database, they can be very efficiently managed, searched, and redefined. Even hyperlinks have attributes.

Each object is labeled with a globally unique object identifier that allows unambiguous referencing from every Hyperwave server. Since these identifiers will not be re-used when the document is replaced or deleted, remote Hyperwave objects can be cached without requiring a sophisticated cache update protocol.

Another aspect of Hyperwave allows multiple servers to be organised into a 'server pool' that shares a common user-hierarchy and keeps the distributed data consistent within the pool (e.g., within a corporate intranet). The Hyperwave protocol is session oriented and stores status details ('session information') about every user currently connected to the system, which can help facilitate various real-time user-to-user communication possibilities.

3 A Simple Recommender System (SRS)

At the University of Auckland's HyperMedia Unit we are prototyping what we term a Simple Recommender System (SRS), based on the fact that we as humans are inclined to express opinions about things. As mentioned in Section 1, the system that we are developing also rests on the assumption that people in a closely collaborating group will actively make recommendations to each other - provided it is reasonably easy to do so. SRS requires that group members use a Shockwave- or Java-enhanced web browser, and that they are logged on to the Unit's local Hyperwave system.

The enhanced browser window will enable users to easily make evaluated and recipient-list-directed recommendations about the currently active page (see Section 3.2). The system will then incorporate all of these recommendation into an interlinked database of URLs, in such a way that each individual user's view is

completely customised on the basis of their own past interaction with both online resources and the rest of the collaborating group.

An absolutely critical aspect of this overall scheme is an intuitive means for users to provide feedback about the accuracy of any recommendations coming through the system, essentially to 'vote' on the relevance of each media resource encountered. This gives the local assessment database a means to adapt its 'rating' of each referee in terms of their success at tagging appropriate resources, from the point of view of the recipient.

3.1 Client-Server Arrangements

3.1.1 Server-side Considerations

On a Hyperwave server the group administrator initialises the SRS by setting up an appropriate collection structure that reflects primary categories of interest to the group. The administrator may also 'seed' the initial collection with URLs of introductory material needed by new users. We'll use Hyperwave's attributes to first prioritise those documents which we consider to be 'core' content; e.g., any documents about using the system itself that new group members should read before beginning to look at other users' recommendations.

In the background will be the software machinery that manages the user-database and meta-collection of recommended links. This can be configured to recognise identical links, having been input by different users, and accept only one numerical evaluation of any particular link from each individual user. Each user can later return at any time, however, to revise their 'value judgment' about a specific online resource. On a regular basis the system will perform a simple statistical analysis of the cumulative assessment of each link and holds the result as a 'group' evaluation attribute. This 'democratically' averaged evaluation can then be viewed within a user's browser interface, whenever they access that resource.

The most critical aspect of the system will be the database analysis functionality that keeps track of the complex pattern of evaluation and interaction between users. It is beyond the scope of this paper to describe the technical details of this process - other than to mention that any resource assumes a relatively higher value within an individual user's own index if it has been specifically recommended, rather than just personally evaluated at the same numerical level. Thus any interpersonal action is rated higher than the same isolated system event.

3.1.2 Client-side Considerations

All of this depends, of course, on users having an effective means of interacting with the background system. To the greatest extent possible, the design of the interface and patterns of interaction must be determined by the needs of end users, rather than by the constraints of any particular suite of software. We have therefore chosen to approaching the production of the client components of the system along two parallel avenues of development: Shockwave and Java.

Shockwave documents are the runtime versions produced from Macromedia's 'Director' application, which require a plug-in to function within a browser. Director is one of the most versatile high-level authoring tools for interactive media, while its underlying language 'Lingo' allows for a reasonable amount of customised programming. As such, initial development with Director/Lingo provides a rapid prototyping environment for implementing various possible forms of SRS client appearance and functionality.

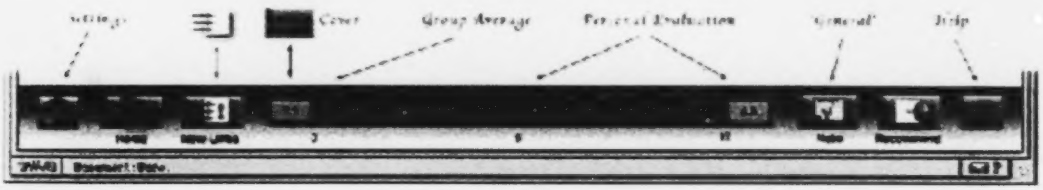
Java, on the other hand, offers the attraction of being able to interlink much more closely with the operation of a web browser and thus more easily access the full capabilities of Hyperwave. This carries the cost however of a longer development cycle and (at present) considerably less assurance of reliable operation on varied systems, compared to an implementation based on a proprietary plug-in.

These two lines of development are not really so separate, since Macromedia is working to position Director as a higher-level programming tool for Java, as well as for its own proprietary formats. Unfortunately, however, the Java output from the current 6.5 version of Director is not yet capable of handling the tasks that are normally delegated to 'helper' components, termed 'Xtras', in the native Director environment. While the functionality provided by these Xtras is not absolutely essential to our initial recommender system, it does

constrain the design to rely only on those network activities that can be delegated to a normal web browser. This seems an acceptable trade-off at this alpha stage of development, though, as a means to readily implement a variety of different approaches to determine the most effective design for the system's basic interface and user interaction.

3.2 Making Recommendations

One of the most important considerations in the design of the SRS is that it should be as easy as possible to make recommendations. Whenever a group member logs onto the group's site, using a Shockwave or Java-enhanced browser, they will see a new-recommendation page (described in Section 3.3.1) along with a specialised interactive options bar as shown in the 'screenshot' below:



The browser-based options bar has the following features (viewed left to right):

- A 'Settings' button which lets each user customise their interface as well as their personal relationship to the overall system.
- A 'Home' button. Clicking on this takes the user to their Home collection archive which serves to index all previously followed recommendations, on a prioritised basis, as well as holding links to that user's own files on the server.
- A 'New Links' button, showing an exclamation point if any fairly highly-valued fresh links are available, which will then bring up an automatically-structured display of all recommendations that have been received since the last (cf. Section 3.3.1), or appears greyed-out if there are no fresh links.
- A central 'Value' slider for entering a personal value judgement about the resource currently displayed. In the case of any document that is new to the user, this slider is preset to a middle value for an 'average' recommendation. (Note that the number scale could be set by each user, to suit their preference for 'granularity' of assessment, and then cross-calibrated at the server.)
- The 'Group' summary field. This is initially a closed window that the user can click to open, revealing the mean value of all evaluations of the current document by other users. This policy of initial concealment was chosen as a way to encourage users to make their own independent evaluations, before effectively polling the average judgment of the whole group.
- A 'General' notation button that sends the URL of the current document to the user's own archive – as well as the group collection - with a value, but without choosing a category for it or designating any specifically defined subset of recipients.
- A 'Specific' recommendation button. On clicking this button the user is provided with a separate interface window wherein they can define the category, user group, and even individual user name(s) for this particular recommendation.
- A 'Help' button, offering context-dependent information about the system.

3.3 Receiving Recommendations

Although the core functionality of the SRS will rely primarily on graded recommendations made directly by users of the system, there are other interesting possibilities. For example, the system can be extended to match up user profiles (based upon their detailed history as evaluators) and then make its 'own' prioritised

BEST COPY AVAILABLE

recommendations. This effectively serves to create identified but 'virtual' referees (based on an automated analysis of users with similar profiles, who have not otherwise specifically exchanged links that appear to be relevant to each other). This is the general sort of recommendation strategy employed by the Firefly system [http://www.firefly.net]

As described in the next three sections, group members will have several different ways of viewing the recommendations that they've received:

3.3.1 The New Links Display

The New Links 'page' will display the recommended URLs as a list sorted on the basis of an analysis of the referral scores for each individual link. The system enables this page to be personalised on the basis of several interrelated factors. For a start, some users may choose to receive only recommendations that fall into their own selected categories. The display graphics themselves can be selected to suit their own representational choices. Then they can adjust the bias of the system's calculations, giving more or less weight to recommendations that come from any other identified user. Finally, they can specify the degree to which they want the system to adaptively readjust the weighting of each referee's new submissions, based on their own history of evaluated reactions to all of that person's previous recommendations.

Just how the New Links page is presented can also vary widely. Although our initial SRS design only displays a sorted list of URLs, a later version of the system will dynamically create a customised newspaper-like HTML document. This can include banner headlines, re-sized graphics, and selected sections of text - all automatically laid out on the basis of the same underlying sorted-list functionality [Fenn & Shearer, 1996].

3.3.2 The User's Home Collection of Recommendations

In a Hyperwave environment, each identified user has a Home collection that is their personal document space, to which they have both read and write access. Within our recommendation system, the server itself can automatically write to this collection, creating and updating a prioritised index of all the recommendations previously followed or freshly introduced by that user.

In the SRS, these collections and links are to be displayed simply as hierarchies within a browser window. However, we believe that ultimately the system could benefit greatly from having an alternate mode of access to this Home data, including a suite of specialised interaction tools, for the manipulation of data where *time*, rather than space, is the essential coordinate. The Lifestreams concept, originally conceived at Yale University, makes very compelling arguments for the further development of such a temporal interface metaphor [http://www.lifestreams.com].

3.3.3 Browsing and Querying the Group Collection

The link to this shared view of all recommended links is to be found at the top of each individual user's Home collection. In essence, it is managed by the system in much the same way as each individual's Home archive of SRS links, except that in this case the evaluation criteria is the averaged judgment of the group as a whole and the structure is maintained by the system administrator. New users in particular may find it useful to be able to browse the structured collections containing past years' recommendations. In addition, all users can query this group database using the Verity search engine that is built into Hyperwave.

4 Maintaining Referential Integrity within a Library of Recommendations

The archive of recommendations can form a most valuable resource library and will become an important part of the group's knowledge base, but one of the most difficult aspects of electronic library maintenance is that of ensuring the referential integrity of the links themselves. Within any really useful system these links can number in the hundreds of thousands and it is not humanly possible for any webmaster to maintain them all. Fortunately Hyperwave, as mentioned in Section 2, maintains the link integrity of all documents stored on a 'pool' of Hyperwave servers [Kappe, 1995]. For links to URLs on conventional web servers, the HyperMedia Unit has developed a program to regularly check these external documents, notify interested parties of any

changes, and supply at least the text of any missing documents from a local cache [Anderson and Lennon, 1998].

5 Conclusion

The system we have described will help to establish a shared knowledge base by way of recommendations provided by participants in online group collaboration. The process relies on identified users who intentionally classify URLs, but then employs a database of past activity and an automated evaluation meta-structure to manage the weighted distribution of recommendations. It can thus effectively track and 'learn' each user's preferences for online media by giving individual documents particular priority with respect to the judgment of other referees - without having to rely on any high-powered AI capabilities. We believe that the further development of this approach to collaborative filtering will yield systems (and related service offerings) of very widespread usefulness - particularly within dedicated intranet settings.

6 References

[Anderson and Lennon, 1998] Anderson, C. and Lennon, J., (1998). Maintaining Link Integrity to External Web Sites in a Hyperwave-based Learning Environment. In proc. *ED-MEDIA 98 - World Conference on Educational Multimedia and Hypermedia*. AACE.

[Andrews et al., 1994] Andrews, K., Kappe, F. and Maurer, H., (1994). Hyper-G and Harmony: Towards the next generation of networked information technology. *Technical report*, IICM.

[Andrews et al., 1995] Andrews, K., Kappe, F., Maurer, H. and Schmaranz, K., (1994). On second generation hypermedia systems. In proc. *ED-MEDIA 95 - World Conference on Educational Multimedia and Hypermedia* pages 75-80. AACE.

[Communications, 1997] Communications. Special Issue on Recommender Systems. *Communications of the ACM*, 40(3).

[Fenn & Shearer, 1996] Fenn, B., Shearer, J., (1996). Delivering 'The Daily Us'. In proc. *WebNet 96 - World Conference of the Web Society*, pages 142-6. AACE

[Flohr, 1995] Flohr, U., (1995). Hyper-G Organizes the Web (1995). *Byte*, November, pages 59-64.

[Kappe, 1995] Kappe, F., (1995). A scalable architecture for maintaining referential integrity in distributed information systems. *Journal of Universal Computer Science*, 1(2), pages 84--104.

[Maurer, 1996] Maurer, H., (1996). *HyperWave: The Next Generation Web Solution*. Addison-Wesley, UK.